second process [having first and second processors] and a server [operatively coupled] over a computer network, the computer program product comprising:

a computer usable medium having program code [means] embodied in the medium [for establishing a point-to-point communications link between the first processor and the second processor over the computer network], the [medium further] program code comprising:

C1

program code for transmitting to the server a network protocol address received by the first process following connection to the computer network;

program code [means] for transmitting, [from the first processor] to the server, a query as to whether the second [processor] process is connected to the computer network;

program code [means] for receiving a network protocol address of the second [processor] process from the server, when the second [processor] process is connected to the computer network; and

program code [means], responsive to the network protocol address of the second [processor] process, for establishing a point-to-point communication link between the first [processor] process and the second [processor] process over the computer network.

C2

2

2~~3~~. (Amended) [A computer server] An apparatus for enabling point-to-point communications between a first and a second [processor] process over a computer network, the [server] apparatus comprising:

a [server] processor;

a network interface [means], operatively coupled to the [server] processor, for connecting the [server] apparatus to the computer network;

a memory, operatively coupled to the processor, for storing a network protocol address for selected of a plurality of [processors connected] processes, each network protocol address stored in the memory following connection of a respective process to the computer network;

means, responsive to a query from the first [processor] process, for determining the on-line status of the second [processor] process and for transmitting [the] a network protocol address of the second [processor] process to the first [processor] process in response to a positive determination of the on-line status of the second [processor] process.

C2

3 ~~24~~. (Amended) The computer server apparatus of claim 2 ~~23~~ further comprising a timer [means], operatively coupled to the [server] processor, for time stamping the network protocol addresses stored in the memory.

4 ~~26~~. (Amended) [In a connection server having a database and a computer network operatively coupled thereto, a] A method for enabling point-to-point communication between a first [processing unit] process and a second [processing unit] process over a computer network, the method comprising the steps of:

C3

A. receiving and storing into a computer memory [storing in the database,] a respective network protocol address for [each] selected of a plurality of [processing units] processes that have an on-line status with respect to the computer network, each of the network protocol addresses received following connection of the respective process to the computer network;

B. receiving a query from the first [processing unit] process to determine the on-line status of the second [processing unit] process;

C. determining the on-line status of the second [processing unit] process; and

D. transmitting an indication of the on-line status of the second [processing unit] process to the first [processing unit] process over the computer network.

5 ~~27~~. (Amended) The method of claim 4 ~~26~~ wherein step C further comprises the steps of:

c.1 searching the [database] computer memory for an entry relating the second [processing unit] process; and

c.2 retrieving [the] a network protocol address of the second [processing unit] process in response to a positive determination of the on-line status of the second [processing unit] process.

C3

~~28~~6. (Amended) The method of claim ~~26~~4 wherein step D further comprises the steps of:

d.1 transmitting the network protocol address of the second [processing unit] process to the first [processing unit] process when the second [processing unit] process is determined in step C to have a positive on-line status with respect to the computer network.

~~29~~7. (Amended) The method of claim ~~26~~4 wherein step D further comprises the steps of:

d.1 generating an off-line message when the second [processing unit] process is determined in step C to have a negative on-line status with respect to the computer network; and

d.2 transmitting the off-line message to the first [processing unit] process.

~~30~~8. (Amended) The method of claim ~~26~~4 further comprising the steps of:

E. receiving an E-mail signal comprising a first network protocol address from the first [processing unit] process; and

F. transmitting the E-mail signal over the computer network to the second [processing unit] process.

~~31~~9. (Amended) The method of claim ~~30~~8 wherein the E-mail signal further comprises a session number and wherein step F further comprises the step of:

f.1 transmitting the session number and network protocol address over the computer network to the second [processor] process.

10 ~~32~~. (Amended) In a computer system, a [A] method for establishing a point-to-point communication link from a caller [processor] process to a callee [processor] process over a computer network, the caller [processor having] process having a user interface and being operatively [coupled] connectable to the callee [processor] process and a server over the computer network, the method comprising the steps of:

C3

A. [generating an] providing a user interface element representing a first communication line;

B. [generating an] providing a user interface element representing a first callee [processor] process; and

C. establishing a point-to-point communication link from the caller [processor] process to the first callee [processor] process, in response to a user associating the element representing the first callee [processor] process with the element representing the first communication line.

11 ~~33~~. (Amended) The method of claim 10 ~~32~~ wherein step C further comprises the steps of:

c.1 querying the server as to the on-line status of the first callee [processor]/ process and

E

c.2 receiving a network protocol address of the first callee [processor] process over the computer network from the server.

12 ~~34~~. (Amended) The method of claim 10 ~~32~~ further comprising the step of:

D. [generating] providing an element representing a second communication line.

13 ~~35~~. (Amended) The method of claim 12 ~~34~~ further comprising the step s of:

E

E. terminating the point-to-point communication link from the caller [processor] process to the first callee [processor] process, in response to the user disassociating the element representing the first callee [processor] process from the element representing the first communication line; and

F. establishing a different point-to-point communication link from the caller [processor] process to the first callee [processor] process, in response to the user associating the element representing the first callee [processor] process with the element representing the second communication line.

14 ~~36~~. (Amended) The method of claim ~~32~~ 10 further comprising the steps of:

D. [generating an] providing a user interface element representing a second callee [processor] process; and

E. establishing a conference point-to-point communication link between the caller [processor] process and the first and second callee [processors] process, in response to the user associating the element representing the second callee [processor] process with the element representing the first communication line.

15 ~~37~~. (Amended) The method of claim ~~32~~ 10 further comprising the step of:

F. removing the second callee [processor] process from the conference point-to-point communication link in response to the user disassociating the element representing the second callee [processor] process from the element representing the first communication line.

16 ~~38~~. (Amended) The method of claim ~~32~~ 10 further comprising the steps of:

D. [generating an] providing a user interface element representing a communication line having a temporarily disabled status; and

E. temporarily disabling a point-to-point communication link between the caller [processor] process and the first callee [processor] process, in response to the user associating the element representing the first callee [processor] process with the element representing the communication line having a temporarily disabled status.

17 ~~39~~. (Amended) The method of claim 16 ~~38~~ wherein the element [generated] provided in step D represents a communication line on hold status.

18 ~~40~~. (Amended) The method of claim 17 ~~39~~ wherein the element [generated] provided in step D represents a communication line on mute status.

C3

19 ~~41~~. (Amended) The method of claim 10 ~~32~~ wherein the caller [processor] process further comprises a visual display and the user interface comprises a graphic user interface.

C4

20 ~~42~~. (Amended) The method of claim 19 ~~41~~ wherein the steps of establishing a point-to-point link as described in step C is performed in response to [a user manipulating] manipulation of the graphic elements on the graphic user interface.

C5

21 ~~43~~. (Amended) A computer program product for use with a computer system comprising:

a computer usable medium having program code [means] embodied in the medium for establishing a point-to-point communication link from a caller [processor] process to a callee [processor] process over a computer network, the caller [processor] process having a user interface and being operatively [coupled] connectable to the callee [processor] process and a server over the computer network, the medium further comprising:

program code [means] for generating an element representing a first communication line;

program code [means] for generating an element representing a first callee [processor] process;

program code [means], responsive to a user associating the element representing the first callee [processor] process with the element representing the first communication line, for establishing a point-to-point communication link from the caller [processor] process to the first callee [processor] process.

22 ~~44~~. (Amended) The computer program product of claim 21 ~~43~~ wherein the program code [means] for establishing a point-to-point communication link further comprises:

program code [means] for querying the server as to the on-line status of the first callee [processor] process; and

program code [means] for receiving a network protocol address of the first callee [processor] process over the computer network from the server.

23 ~~45~~. (Amended) A computer program product of claim 21 ~~43~~ further comprising:

program code [means] for generating an element representing a second communication line.

24 ~~46~~. (Amended) The computer program product of claim 23 ~~45~~ further comprising:

program code [means], responsive to the user disassociating the element representing the first callee [processor] process from the element representing the first communication line, for terminating the point-to-point communication link from the caller [processor] process to the first callee [processor] process; and

program code [means], responsive to the user associating the element representing the first callee [processor] process with the element presenting the second communication line, for establishing a different point-to-point communication link from the caller [processor] process to the first callee [processor] process.

25 ~~47~~. (Amended) The computer program product of claim 21 ~~43~~ further comprising:

program code [means] for generating an element representing a second callee [processor] process; and

program code means, responsive to the user associating the element representing the second callee [processor] process with the element representing the first communication line, for establishing a conference communication link between the caller [processor] process and the first and second callee [processors] process.

26 ~~48~~. (Amended) The computer program product of claim 25 ~~47~~ further comprising:

program code [means], responsive to the user disassociating the element representing the second callee [processor] process from the element representing the first communication line, for removing the second callee [processor] process from the conference communication link.

27 ~~49~~. (Amended) The computer program product of claim 21 ~~43~~ further comprising:

C5

program code [means] for generating an element representing a communication line having a temporarily disabled status; and

program code [means], responsive [to user associating] association of the element representing the first callee [processor] process with the element representing the communication line having a temporarily disabled status, for temporarily disabling the point-to-point communication link between the caller [processor] process and the first callee [processor] process.

28 ~~50~~. The computer program product of claim 27 ~~49~~ wherein the communication line having a temporarily disabled status comprises a communication line on hold status.

29 ~~51~~. The computer program product of claim 27 ~~49~~ wherein the communication line having a temporarily disabled status comprises a communication line on mute status.

30 ~~52~~. (Amended) A computer program product of claim 21 ~~43~~ wherein the computer system [caller processor] further comprises a visual display and the user interface comprises a graphic user interface.

31 ~~53~~. (Amended) The computer program product of claim 30 ~~52~~ wherein the element representing the first communication line and the element representing the first callee [processor] process are graphic elements and wherein the

40

program code [means] for establishing a point-to-point communication link from the caller [processor] process to the first callee [processor] process further comprises:

C5 program code [means], responsive to [a user manipulating] manipulation of the graphic elements on the graphic user interface, for establishing the point-to-point communication link from the caller [processor] process to the first callee [processor] process.

32 ~~54~~. (Amended) A method of locating a [user] process over a computer network comprising the steps of :

C6 a. maintaining an Internet accessible list having a plurality of selected entries, each entry comprising an [electronic mail address] identifier and a corresponding Internet protocol address [for] of a process currently connected to the Internet, the Internet Protocol address added to the list following connection of the process to the computer network; and

b. in response to identification of one of the list entries by a requesting process, providing one of the [electronic mail address] identifier and the corresponding Internet protocol address of the identified entry to the requesting process.

33 ~~55~~. (Amended) A method for locating [users] processes having dynamically assigned network protocol addresses over a computer network, the method comprising the steps of:

a. maintaining, in a computer memory, a network accessible compilation of entries, [each entry] selected of the entries comprising a network protocol address and a corresponding identifier [for a user] of a process connected to the computer network[;], the network protocol address of the corresponding process assigned to the process upon connection to the computer network; and

b. in response to identification of one of the entries by a requesting process providing one of the identifier and the network protocol address to the requesting process.

34 ~~56~~. (Amended) The method of claim ~~55~~ 33 further comprising the step of:

c. modifying the compilation of entries.

35 ~~57~~. (Amended) The method of claim ~~56~~ 34 wherein step c further comprises:

c.1 adding an entry to the compilation upon the occurrence of a predetermined event.

36 ~~58~~. (Amended) The method of claim ~~57~~ 35 wherein the predetermined event comprises notification by a user process of an assigned network protocol address.

37 ~~59~~. (Amended) The method of claim ~~56~~ 34 wherein step c further comprises:

c.1 deleting an entry from the compilation upon the occurrence of a predetermined event.

38 ~~60~~. (Amended) A computer program product for use with a [server apparatus] computer system having a memory and being operatively [coupled] connectable over a computer network to one or more computer processes, the computer program product comprising a computer usable medium having program code embodied in the medium the program code comprising:

a. program code configured to maintain, in [a] the computer memory, a network accessible compilation of entries, [each entry] selected of the entries comprising a network protocol address and a corresponding identifier [for] of a process connected to the computer network, the network protocol address of the corresponding process assigned to the process upon connection to the computer network; and

b. program code responsive to identification of one of the entries by a requesting process and configured to provide one of the identifier and the network protocol address to the requesting process.

39 ~~61~~. (Amended) The computer program product of claim ~~60~~ 38 further comprising:

c. program code configured to modify the compilation of entries.

C6

40 ~~62~~. (Amended) The computer program product of claim ~~61~~ 39 wherein program code configured to modify comprises:

c.1 program code configured to add an entry to the compilation upon the occurrence of a predetermined event.

41 ~~63~~. (Amended) The computer program product of claim ~~62~~ 40 wherein the predetermined event comprises notification by a process of an assigned network protocol address.

42 ~~64~~. (Amended) The computer program product of claim ~~60~~ 38 wherein step c further comprises:

c.1 program code configured to delete an entry from the compilation upon the occurrence of a predetermined event.

C7

43 ~~66~~. (Amended) A computer program product for use with a computer system, the computer system [including] executing a first process operatively coupled over a computer network to a second process and a server process , the computer program product comprising a computer usable medium having computer readable program code embodied therein, the program code [means] comprising:

a. program code configured to access a directory database, the database having a network protocol address for a selected plurality of processes having on-line status with respect to the computer network, the network protocol address of each respective process forwarded to the database following connection to the computer network; and

b. program code responsive to one of the network protocol addresses and configured to establish a point-to-point communication link from the first process to the second process over the computer network.

44

~~67~~. (Amended) In a first computer process operatively coupled over a computer network to a second process and an address server, a method of establishing a point-to-point communication between the first and second processes comprising the steps of:

A. following connection of the first process to the computer network forwarding to the address server a network protocol address at which the first process is connected to the computer network;

[A.] B. querying the address server as to whether the second process is connected to the computer network;

[B.] C. receiving a network protocol address of the second process from the address server, when the second process is connected to the computer network; and

[C.] D. in [responsive] response to the network protocol address of the second process, establishing a point-to-point communication link with the second process over the computer network.

Remarks

Applicants have considered carefully the Office Action dated October 28, 1998 and the references cited therein. In response, the claims have been amended. Applicants respectfully request reexamination and reconsideration of the application.

Claims 1-4, 6-11, 21, 23-24, 26-64, 66 and 67 have been examined and are rejected over various combinations of U.S. Patent 5,608,786(Gordon); U.S. Patent 5,740,231 (Cohn); U.S. Patent 5,524,254 (Morgan); and excerpts from The World Wide Web Unleashed (December). Before responding to the

44